MCP SERVER | NO CODE | CLOUD HOSTED

Deterministic Global Postal Formatter MCP for AI Agents

Standardizing International Address Formats for Data Pipelines

The Deterministic Global Postal Formatter automatically validates and standardizes international postal codes for any data pipeline. It enforces strict, region-specific rules—like adding hyphens to Brazilian CEPs or spaces to UK Postcodes—ensuring your address records are always clean and compliant.

F Quality Score 9.37/100

regex | data-validation | address-formatting | crm-data | localization

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · JUNE 2026
vinkius.com/mcp/deterministic-global-postal-formatter

SCAN TO VIEW

The connectivity layer between AI and the world's software.

Vinkius sits between AI and every application. All communication passes through Vinkius Cloud via the Model Context Protocol (MCP) — with governance, observability, and security at every layer.

Your AI Connections Run Through Vinkius Cloud

The world's largest managed MCP catalog

Vinkius is the connectivity layer where AI connects to the software your business already runs. We handle the hosting, the security, the credentials, the uptime — you get agents that actually do things.

We operate the world's largest managed MCP catalog. Major SaaS platforms, CRMs, databases, and cloud providers — running, monitored, production-ready. This MCP server is hosted and maintained by the Vinkius Cloud for AI Agents.

> *The agent doesn't manage credentials, doesn't manage uptime, doesn't manage security. Vinkius does.*
>
> — Architecture principle

Four Pillars of the Vinkius Runtime

01 — Security by design

Credentials stay encrypted at rest via AES-256. The AI agent never touches raw keys — they're injected into a sandboxed V8 isolate at runtime. Actions are logged, and connections have an emergency kill switch.

02 — Built on MCP Fusion

This MCP server was built with **MCP Fusion**, the open-source framework (Apache 2.0) that powers the entire Vinkius catalog. Schema-as-firewall strips undeclared fields, compiled PII redaction runs at zero overhead, and cryptographic lockfiles produce git-diffable audit trails.

03 — Deterministic observability

Eight immutable metrics per endpoint: request volume, p95 latency, error rate, active connections, cost attribution. A live payload feed logs every tool call with mutation detection.

04 — Autonomous operations

Servers are deployed, monitored, and patched autonomously. New capabilities and security patches ship weekly. Zero-downtime deployments ensure continuous availability across all managed MCP servers.

AES-256	Ed25519	24h TTL	V8 Isolate
Encryption at rest	PKI vault signatures	Ephemeral session keys	Sandboxed execution

One Token. Instant Access.

Every MCP server on Vinkius is accessed through a **Connection Token**. Tokens are generated in the cloud dashboard and produce a unique MCP endpoint URL. Paste this URL into any MCP-compatible client — no SDK required.

A single token can serve **multiple AI clients simultaneously**, or you can issue separate tokens per client for granular access control. Each token tracks its own request count, last activity timestamp, and can be individually enabled or revoked.

MCP ENDPOINT

```
https://edge.vinkius.com/{token}/mcp
```

Claude · Cursor · VS Code · Windsurf · Grok · Gemini

Security Is the Architecture

Security in Vinkius is not a feature — it's the foundation of the runtime. The gateway enforces multiple independent protection layers between AI agents and third-party APIs.

01 — Ed25519 PKI Vault

Every workspace has an Ed25519 Master Key. Session keys are generated ephemerally (24h TTL) and signed by the Master Key. Credentials never leave the vault boundary.

02 — V8 Isolate Sandboxing

Tool code runs inside isolated-vm V8 isolates with 64 MB memory caps and per-request timeouts. No filesystem access, no network access except through the SSRF-guarded fetch bridge.

03 — SSRF Guard

All outbound HTTP requests are DNS-resolved and validated before execution. Private IP ranges (10.x, 172.16-31.x, 192.168.x, AWS metadata 169.254.x) are blocked at the network layer.

04 — DLP & PII Redaction

A ResponseGuard pipeline intercepts every tool response. Configurable redaction patterns strip sensitive fields (emails, SSNs, card numbers) before data reaches the AI agent.

05 — Cryptographic Audit Trail

Every request is signed into a SHA-256 hash chain with Ed25519 signatures. Events form a tamper-proof, SIEM-exportable forensic record.

06 — Honeytoken Trap System

Phantom credentials are injected into isolated environments. If a honeytoken is used outside Vinkius infrastructure, the server is quarantined instantly.

Emergency Kill Switch

EU AI Act Art. 14(1) Compliant

The kill switch is an **emergency halt** mechanism — not a simple toggle. When triggered, it executes three actions atomically:

01 — Server deactivated

The MCP server is immediately taken offline across the entire cluster.

02 — All tokens revoked

Every connection token is invalidated. Total lockout — reconnection blocked until new tokens are issued.

03 — WebSocket connections killed

Active connections terminated via Redis pubsub broadcast. Propagates to every runtime node in the cluster.

Full Visibility. Zero Guesswork.

The Vinkius cloud dashboard includes a full MCP Governance suite — real-time analytics and security controls for production AI operations.

Control Plane

KPI dashboard with request volume, latency, success rate, token consumption, and AI-generated operational briefings.

Agent Activity

Which AI clients are connecting, how often, and what they're doing — real-time session tracking.

FinOps

Cost tracking per tool, payload compression savings, budget optimization signals, and consumption trends.

Tool Health

Slowest and most error-prone tools, with actionable root-cause insights and performance baselines.

Firewall & DLP

PII redaction activity, sensitive data protection counters, and security event timeline.

Incident Log

Error trends, failure rates, status-code breakdowns, and forensic audit trail access.

> Get started at **cloud.vinkius.com** — connect your AI agent in under 60 seconds.

Deterministic Global Postal Formatter MCP

1 tools available
Cloud-hosted on Vinkius

Messy customer data is a huge drag on any system. When you're dealing with international addresses, regional quirks kill accuracy. A simple AI model might guess at a ZIP code but struggle when it comes to knowing that a Brazilian CEP needs a hyphen (`01311-000`) or that the UK Postcode requires a strategic space (`SW1A 1AA`). This MCP solves those structural problems by running validation through strict, deterministic regex engines. It doesn't guess; it checks against known global rulesets for Brazil, the US, the UK, and Canada.

It takes raw inputs—whether they are missing hyphens or just run together—and automatically formats them perfectly according to local compliance standards. This makes your data immediately usable in CRMs or billing systems without manual cleanup. Since Vinkius hosts this MCP as part of its catalog, you connect it once from any AI client and gain access to reliable address standardization for all your global projects.

Core Capabilities

01 — Standardize International Postal Formats

It takes raw postal codes (like 'sw1a1aa') and instantly reformats them into the correct, compliant structure ('SW1A 1AA').

02 — Validate Structural Integrity

The MCP rejects addresses that are structurally impossible for their region, providing a clear reason why the data fails validation.

03 — Handle Multi-Country Data Sets

It applies distinct rulesets for four major international regions: Brazil (CEP), United States (ZIP/ZIP+4), United Kingdom, and Canada.

One Click on Vinkius — From Prompt to Execution

Available at **vinkius.com/mcp/deterministic-global-postal-formatter** — connect your AI agent in three steps.

01 You provide the MCP with a raw postal code string and its corresponding two-letter country code.

02 The Deterministic Global Postal Formatter executes the input against strict, deterministic regex rulesets for that specific country.

03 It returns either the perfectly formatted, validated address code or a detailed report explaining exactly why the original input failed validation.

The bottom line is you get guaranteed compliance. Instead of just guessing an format, your AI client gets back data that's been mathematically verified against global postal standards.

Built For

This MCP is essential for anyone running data pipelines that touch international customer records. It helps Data Ops Engineers who are tired of manually scrubbing address databases, and CRM Managers needing guaranteed clean input before import.

Data Operations Engineer

You use this to build pre-processing steps for data ingestion pipelines, making sure that international address fields never break your ETL processes.

CRM Administrator

You connect this to clean up imported leads or existing customer records, guaranteeing that every postal code is structured correctly before it enters the sales team's view.

Global E-commerce Manager

You process large batches of order data from various regions, ensuring correct addressing for shipping labels and payment verification.

What Changes When You Connect

01 Eliminate structural errors: Instead of having to manually spot which ZIP codes are wrong, the format_postal_code tool instantly flags invalid data and tells you why.

02 Guaranteed compliance: Don't worry about regional differences again. The MCP enforces local standards for everything from Brazilian CEP hyphens to UK spacing.

03 Saves time on cleanup: It handles messy inputs—like codes run together (`01311000`)—and automatically cleans them into perfectly formatted strings, saving hours of manual data scrubbing.

04 Faster pipelines: Because it uses pure JS runtime execution, the formatting happens in microseconds without relying on massive external geolocation databases.

05 Robust validation: You get more than just a pass/fail. The system returns clear diagnostic reports detailing exactly what's wrong with your address component.

Real-World Applications

Cleaning up imported lead lists

A sales operations team imports 10,000 new leads from a global source. Instead of having to write complex regex rules for every country, they run the entire list through the Deterministic Global Postal Formatter MCP using format_postal_code. The result is a clean dataset where all postal codes are standardized and ready for immediate CRM entry.

Verifying shipping addresses before checkout

An e-commerce platform needs to validate the customer's entered address right at the payment stage. They call the MCP, which quickly checks if a US ZIP code is 5 or 9 digits and validates that UK postcode structure, preventing failed shipments.

Migrating historical CRM data

A company acquires another business with decades of dirty customer records. They use the MCP to systematically run every postal code through format_postal_code (e.g., for 'BR' or 'US') before migrating, ensuring the new system doesn't inherit old formatting errors.

Processing international payroll data

The HR team receives salary records from multiple countries. They use the MCP to validate the associated regional codes (like CEP for Brazil) against strict rules, ensuring that all location identifiers are correct before payment processing begins.

Patterns to Avoid

Relying on general AI models alone

✗ AVOID

Asking a generic AI client to simply 'fix the address' without specific regional rules. It might correctly identify a US ZIP but fail completely on Brazilian CEP hyphenation or UK spacing.

✓ INSTEAD

Use the Deterministic Global Postal Formatter MCP and specifically call format_postal_code, providing both the raw code AND the two-letter country code (e.g., 'BR' for Brazil). This forces the AI agent to use deterministic, proven logic.

Hardcoding complex regex rules

✗ AVOID

Writing thousands of lines of custom JavaScript or Python regex functions to handle every international exception. This code becomes massive, difficult to maintain, and hard to debug.

✓ INSTEAD

Connect this MCP directly into your workflow. It abstracts the complexity; you just pass the data and let the validated rules engine handle the heavy lifting.

Assuming all postal codes are simple strings

✗ AVOID

Treating a UK postcode like a simple 6-character code, which fails when the agent doesn't account for mandatory spacing or letter placement.

✓ INSTEAD

Use format_postal_code and ensure you pass the correct country parameter (e.g., 'GB'). The MCP knows that the rules for Great Britain are different from those for Canada.

The Right Fit

You should use this MCP if your process involves ingesting, storing, or sending international address data across multiple countries. It's your definitive source of truth for postal code formatting. However, don't rely on it to solve all address problems. For instance, while it

handles the *postal code* structure (like ensuring 'SW1A 1AA'), it won't validate if a street name actually exists or if a city matches that postcode; you still need other tools for full geographical validation. If your primary pain point is inconsistent formatting and structural errors in postal codes, this MCP is non-negotiable.

Deterministic Global Postal Formatter: Fixing Dirty International Address Data

Today, address cleanup means navigating multiple spreadsheets, cross-referencing country-specific formatting guides, and manually fixing hyphens or spaces. Your team spends hours in a painful loop of copy-pasting data into different validation tools just to make sure a simple CEP or postcode looks right.

With this MCP, your agent handles the whole thing instantly. You feed it the raw address components, and it spits out clean, standardized postal codes that are guaranteed compliant for Brazil, the US, UK, and Canada. It's instant data quality.

Deterministic Global Postal Formatter: Improving Data Integrity Across Regions

The biggest time sink is the uncertainty. You can’t assume that a 5-digit US ZIP code means anything if the source data sometimes gives you 4 digits, or if it's missing the required plus-four extension. This ambiguity slows down every downstream process.

Now, you get certainty. The MCP uses strict rules to not only format but also validate the *structure* of your input. It guarantees that when an agent returns a formatted code, you can trust its integrity for billing or routing.

Deterministic Global Postal Formatter: 1 Tool for International Address Validation

This single tool lets you programmatically check and standardize any global postal code against strict, regional formatting rules.

#	TOOL	DESCRIPTION
01	`format_postal_code`	Validates and reformats global postal codes by accepting a raw code string and an ISO 2-letter country code to ensure compliance with local standards.

See It in Action

Real prompts you can use once this MCP is connected to your AI agent through Vinkius Cloud.

U I need to fix 50 customer records. I've got a mix of US ZIP codes and Brazilian CEPs, all mixed up.

Here's the cleanup report:

- **Record 1 (US):** Input: `90210`. Status: Valid.
- **Record 2 (BR):** Input: `1311000`. Status: Valid. Formatted to `01311-000`.
- **Record 3 (UK):** Input: `sw1a1aa`. Status: Valid. Formatted to `SW1A 1AA`.
- **Record 4 (US):** Input: `123`. Status: Invalid. US ZIP requires at least 5 digits.

U Does 'CA90210' look like a valid address component for Canada?

Validation Check: Canadian Postcode

- Input Code: `CA90210`
- Country Detected: CA (Canada)
- Structural Validation: Failed.
- Reasoning: The format requires a space separating the letters and numbers, and this input is missing that structure. Please try formatting it as 'V5K 0A1'.

U I have a bunch of old data with inconsistent postal codes for Brazil. Can you standardize them?

Brazilian CEP Standardization Report

Please provide the raw code and confirm country='BR'. Here are some examples:

- Raw: `01311000` → **Formatted:** `01311-000` (Success)
- Raw: `75010123` → **Formatted:** `75010-123` (Success)
- Raw: `AABBCCDD` → **Validation Error:** CEP must be 8 digits long.

Frequently Asked Questions

01 **How can I use the Deterministic Global Postal Formatter to clean up my global customer list?**

You run your data through this MCP by providing the raw postal code and its country. It automatically validates every entry against strict rules, guaranteeing that all hyphens, spaces, and digit counts are correct before you save or export the cleaned data.

02 **Does Deterministic Global Postal Formatter handle US ZIP+4 codes?**

Yes. It supports both standard 5-digit US ZIPs and the full 9-digit format (ZIP+4). If the input is structurally invalid, it will report that failure to you.

03 **Is this MCP reliable for multiple countries like Brazil and the UK?**

It's built specifically for reliability. The system uses separate rulesets for each country—for example, knowing the difference between a Brazilian CEP structure and a UK Postcode spacing—making it highly accurate across regions.

04 **What kind of data is too messy for Deterministic Global Postal Formatter?**

It struggles if you mix country codes or provide ambiguous input. You must always give the MCP both the raw postal code and the correct two-letter ISO country code to ensure accurate processing.

05 **Can I use this formatting tool in my existing AI agent workflows?**

Absolutely. By connecting it via Vinkius, your AI client can call the MCP as a function. This means you write a simple instruction like 'Format this address' and let the MCP handle all the complex logic behind the scenes.

Go Live in 60 Seconds

Get your connection token from **cloud.vinkius.com**, then paste the endpoint URL into any MCP-compatible client.

YOUR MCP ENDPOINT

```
https://edge.vinkius.com/[TOKEN]/mcp
```

CLIENT	WHERE TO CONFIGURE
Claude AI	Profile → Customize → Connectors → "+" → Add custom connector → Paste endpoint
Cursor	Settings → Features → MCP Servers → "+ Add New MCP Server" → Type: SSE → Paste endpoint
VS Code	Ctrl/Cmd+Shift+P → "MCP: Add Server" → add `"deterministic-global-postal-formatter": { "url": "…" }`
Windsurf	MCP Settings → `mcp_settings.json` → Add endpoint URL
ChatGPT	Settings → Tools & plugins → Add MCP server → Paste endpoint
Gemini	Extensions → Add MCP Server → Paste endpoint URL

ASK AN AI ABOUT THIS

Let your preferred AI explain this MCP server

- Ask ChatGPT
- Ask Claude
- Ask Perplexity
- Ask Gemini
- Ask Grok

READY TO CONNECT

Deterministic Global Postal Formatter is live on Vinkius Cloud.

Get your connection token, paste it into your AI agent, and start building. No SDK. No deployment. Just results.

Start at cloud.vinkius.com →

vinkius.com · support@vinkius.com

INDEPENDENT PLATFORM DISCLAIMER

Vinkius is an independent platform and is not affiliated with, endorsed by, sponsored by, verified by, or otherwise authorized by Deterministic Global Postal Formatter. All third-party trademarks, logos, and brand names are the property of their respective owners. Their use in this document is strictly for informational purposes to identify service compatibility and interoperability.

DOCUMENT INFORMATION

Generated	June 2026
MCP Server	Deterministic Global Postal Formatter MCP
Server ID	`019e38da-600f-7140-a96f-c9dcb0d949f2`
Platform	Vinkius Cloud for AI Agents
Endpoint	`https://edge.vinkius.com/{token}/mcp`

LICENSE & USAGE

This document is generated automatically by the Vinkius PDF Engine. Content reflects the MCP server configuration at the time of generation and may change as updates are deployed. For the most current information, visit vinkius.com/mcp/deterministic-global-postal-formatter.